

ПРОГРАММНАЯ РЕАЛИЗАЦИЯ АЛГОРИТМОВ ВЫЧИСЛЕНИЯ КОНТРОЛЬНОЙ СУММЫ CRC8 НА МИКРОКОНТРОЛЛЕРЕ ATTINY44

Е. А. Мыцко

Томский политехнический университет
evgenvt@tpu.ru

Введение

В системах промышленной автоматизации, автоматизированных системах управления широко применяются микроконтроллерные системы для сбора и первичной обработки данных с датчиков, а также выработки управляющих сигналов на исполнительные устройства. В таких системах нижний уровень системы может включать множество недорогих микроконтроллерных решений с малым объёмом памяти программ и данных, а также низкой производительностью. Помимо решения основных задач необходимо также обеспечивать приём, подготовку и отправку данных в различные модули микроконтроллерной системы, при этом необходимо обеспечивать защиту передаваемой и обрабатываемой информации от ошибок. Одним из способов защиты передаваемой информации от искажений является применение контрольных сумм CRC. В работе рассмотрены программные реализации алгоритмов вычисления контрольной суммы CRC8 на примере системы измерения температуры с микроконтроллером ATtiny44.

Описание системы измерения температуры

Микроконтроллерная система измерения температуры построена на основе цифрового 1-wire датчика DS18B20 фирмы DALLAS [1] и 8-рядного микроконтроллера ATtiny44 от Atmel. На рисунке 1 представлена структурная схема информационного обмена микроконтроллера с датчиком, включая блоки отправки и приёма данных.

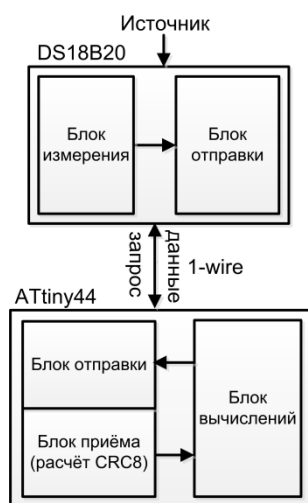


Рис. 1. Система измерения температуры на ATtiny44

Ведущее устройство (микроконтроллер) инициирует запрос на передачу данных с датчика путём отправки на него 64-х разрядного кода

устройства с контрольной суммой CRC8. Датчик в ответ на запрос отправляет пакет данных, содержащий значение температуры, граничные контрольные значения, регистр конфигурации и вычисленную контрольную сумму CRC8.

Существуют различные алгоритмы расчёта контрольной суммы CRC, такие как классический [2], табличный [2] и матричный [3,4]. Реализация классического алгоритма расчета контрольной суммы CRC с побитовым сдвигом требует большое количество тактов процессора и имеет низкое быстродействие. Реализация табличного алгоритма имеет более высокое быстродействие, однако для расчёта 1-го байта контрольной суммы требуется 256 байт памяти для хранения таблицы предвычисленных значений. Реализация матричного алгоритма является некоторым компромиссом между табличной и классической реализации по требованию к объёму памяти и количеству тактов процессора.

Особенности матричного алгоритма вычисления CRC8

На рисунке 2 представлена общая схема вычисления CRC8 на основе матричного алгоритма.

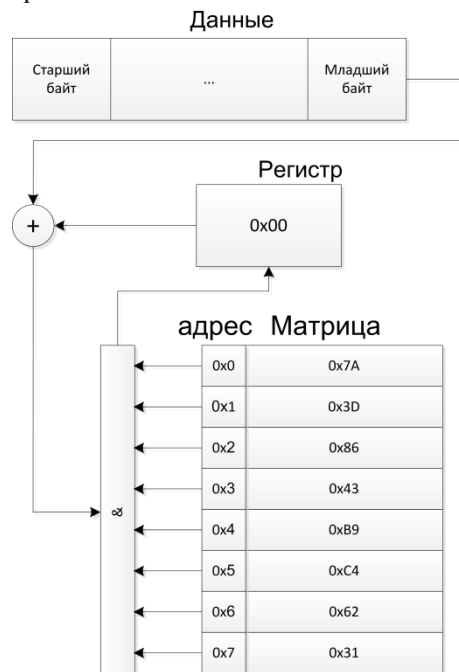


Рис. 2. Вычисление CRC8 матричным алгоритмом

Матрица (рис. 2), в отличие от известного табличного алгоритма [2], требует 8 байт памяти данных и вычисляется путём деления по модулю два [5] полиномов от X^8 до X^{15} на образующий полином $X^8 + X^5 + X^4 + 1$. Сама процедура вычисления

CRC заключается в умножении по модулю 2 байта информации на вычисленную матрицу.

Сравнение различных реализаций вычисления CRC8

Для выбора рекомендации по применению алгоритмов вычисления CRC8 в микроконтроллерных системах осуществлены программные реализации классического, табличного и матричного алгоритмов вычисления контрольной суммы CRC8 на микроконтроллере ATtiny44 для блока данных размером 56 бит. Результаты программных реализаций представлены в таблице, включающей следующие поля:

- «такты» – количество тактов микроконтроллера, необходимых для завершения работы алгоритма,
- «слова» – объём кода (набора команд) программной реализации в словах (слово равно двум байтам),
- «память» объём памяти в словах, необходимый для хранения предвычисленных значений,
- «РОН» – количество регистров общего назначения, применяемых в реализации алгоритма.

Таблица. Сравнение программных реализаций CRC8

Алгоритм	Такты	Слова	Память	РОН
Классический	5516	35	-	5
Табличный	116	16	128	7
Матричный (EEPROM)	634	23	4	6
Матричный (PROM)	603	24	4	10
Матричный (RAM)	545	22	4	10
Матричный (РОН)	244	50	-	13
Матричный (память прог.)	254	42	-	6

По представленным в таблице результатам можно выбрать наиболее подходящую реализацию алгоритма, исходя из доступных ресурсов и требований к быстродействию. Наиболее быстродействующей является табличная реализация, однако для неё требуется 144 слова и 8 регистров. Классический алгоритм является самым медленным (5516 тактов) и требует для реализации 35 слов. Варианты матричного алгоритма в данном случае являются некоторым компромиссом между быстродействием и количеством слов. В случае если увеличение количества слов до 42–50 для модуля расчёта CRC не критично, то наиболее подходящей может быть реализация матричного алгоритма с матрицей в РОН (244 такта) или в памяти программ (254 такта). При значительном

ограничении ресурсов для модуля расчёта CRC можно применить реализацию с матрицей в RAM (22 слова памяти программ и 4 слова RAM), при этом количество тактов относительно реализации на РОН увеличится с 244 до 545.

Заключение

Проведено исследование и сравнение реализаций алгоритмов вычисления CRC8 по быстродействию и требуемому объёму памяти (слов), на основе которого сформированы рекомендации по применению алгоритмов в зависимости от условий реализации. Установлено, что в случае наличия свободных ресурсов памяти под модуль расчёта CRC объёмом от 144 слов, целесообразно применять табличную реализацию в виду её быстродействия. Однако в большинстве случаев для систем измерения на основе микроконтроллеров ATtiny44 основные ресурсы задействованы на выполнения задач, согласно назначению системы и ресурсы для реализации дополнительных модулей контроля целостности значительно ограничены. В таких случаях целесообразно применение реализаций с матрицей в RAM или в регистрах общего назначения.

Список использованных источников

1. DS18B20 – Датчик температуры с интерфейсом 1-Wire [Электронный ресурс]. – URL: <http://mypractic.ru/ds18b20-datchik-temperature-s-interfejsom-1-wire-opisanie-na-russkom-yazyke.html> (дата обращения: 01.10.2017)
2. A painless guide to CRC error detection algorithms [Электронный ресурс]. – URL: http://www.zlib.net/crc_v3.txt (дата обращения: 01.10.2017)
3. Мальчуков А.Н., Осокин А.Н. Быстрое вычисление контрольной суммы CRC: таблица против матрицы // Прикладная информатика. – 2010. – № 2(26). – С. 58–63.
4. Мыцко Е.А., Мальчуков А.Н. Исследование программных реализаций алгоритмов вычисления CRC совместных с PKZIP, WINRAR, ETHERNET // Известия Томского политехнического университета. – 2013 – Т. 322 – № 5. – С. 170-175.
5. Буркатовская Ю.Б., Мальчуков А.Н., Осокин А.Н. Быстродействующие алгоритмы деления полиномов в арифметике по модулю два // Известия Томского политехнического университета. – 2006. – Т. 309. – № 1. – С. 19-24.